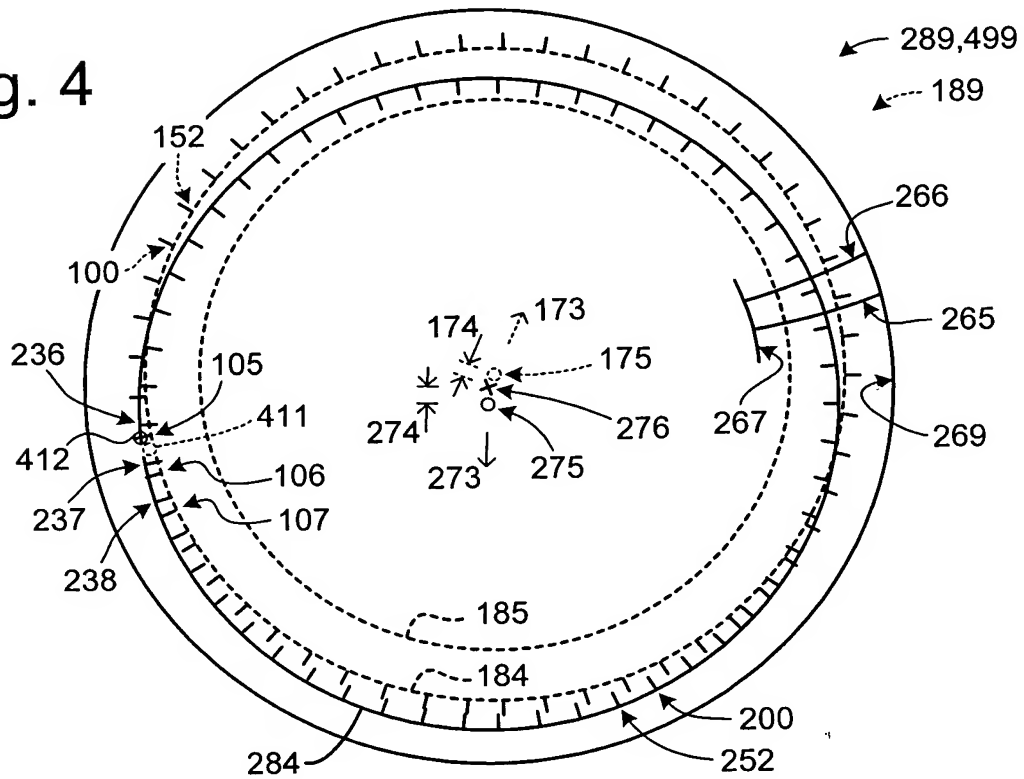


Fig. 3

Fig. 4



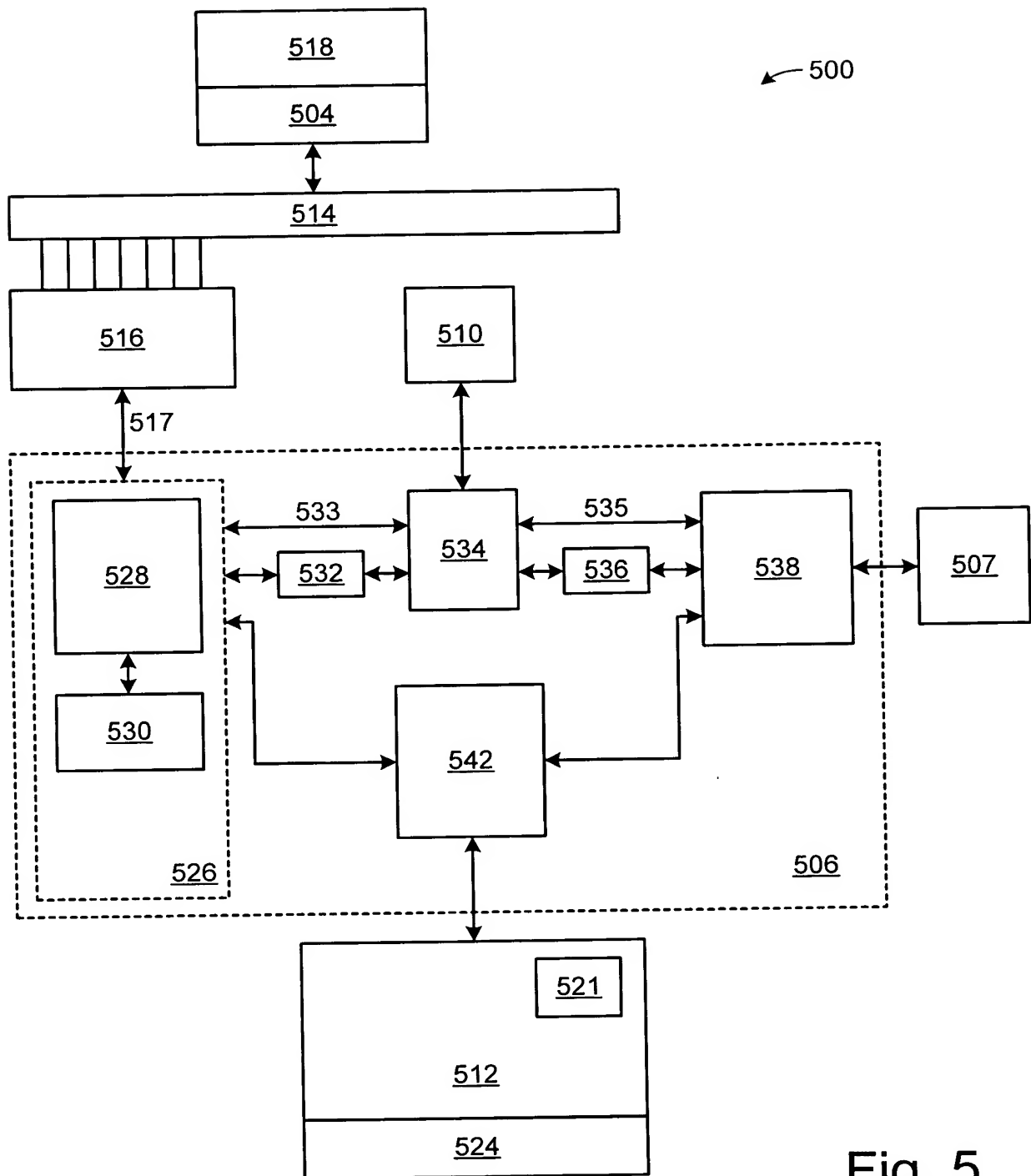


Fig. 5

Fig. 6

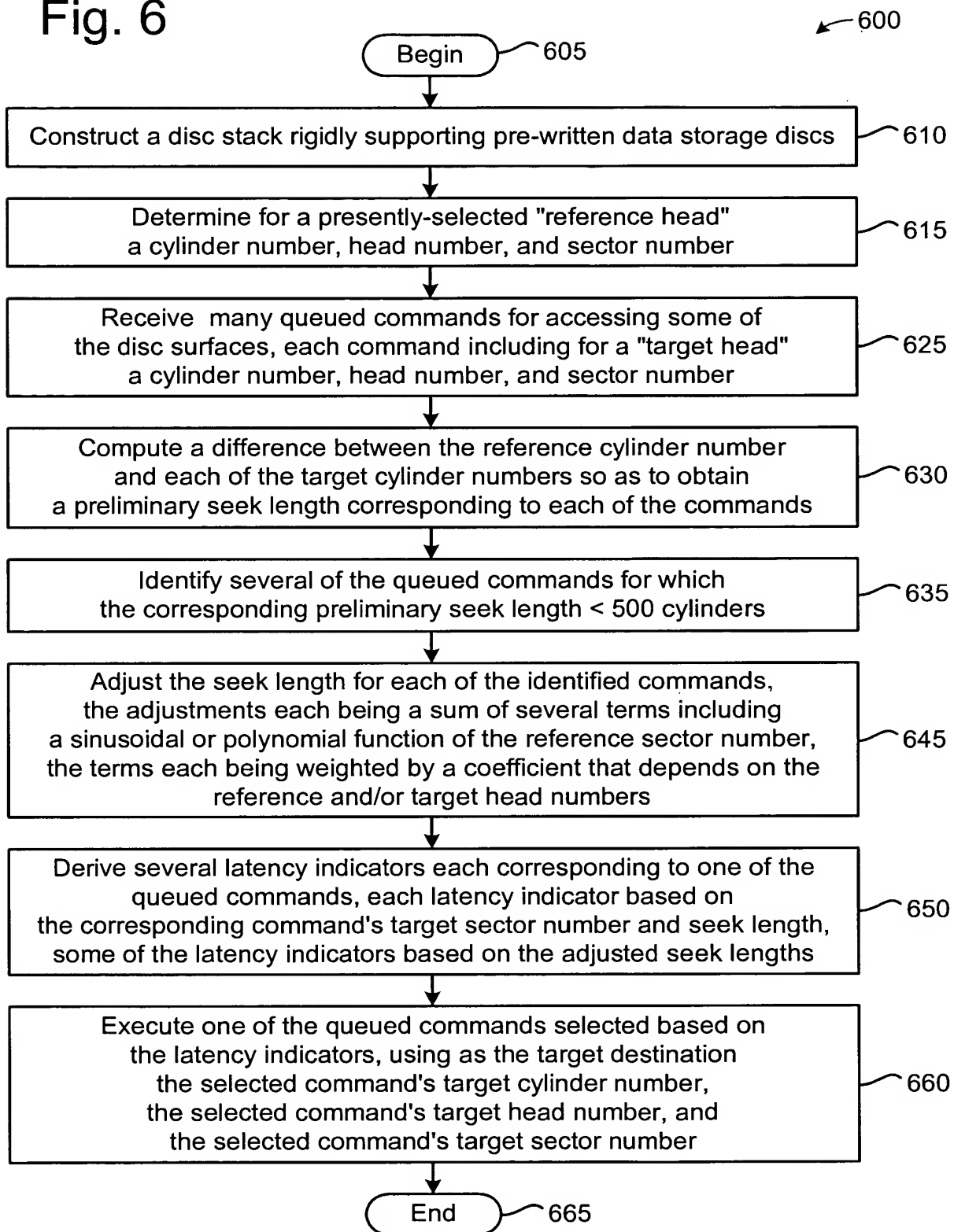


Fig. 7

700



```

// Function: sa_EvalDeltaS ()
// Parameters: S_Head, D_Head, StartSector, SrcCyl
// Return: Adjustment to be added to the preliminary seek length
//=====
//      EffectiveDistance = DestCyl + DeltaL(Dest) - SrcCyl - DeltaL(Src)
//                      = DestCyl + DeltaS(Dest,Src) - SrcCyl
//      DeltaL(Location) = c0(h) + c1(h)*NormalizedSrcCyl>>15
//                      + c2(h)*(NormalizedSrcCyl^2)>>30
//                      + c3(h)*sine(StartingSector) + c4(h)*cosine(StartingSector)
//      DeltaS(Dest,Src) = DeltaL(Dest) - DeltaL(Src)
//                      = c0(D_Head) - c0(S_Head)
//                      + (c1(D_Head) - c1(S_Head)) * NormalizedSrcCyl>>15
//                      + (c2(D_Head) - c2(S_Head)) * (NormalizedSrcCyl^2)>>30
//                      + (c3(D_Head) - c3(S_Head)) * sine(StartingSector)>>15
//                      + (c4(D_Head) - c4(S_Head)) * cosine(StartingSector)>>15
//
//      pd_Opr_ai16Q15SineTable(x) = round(32767*sine(x*2*Pi/SECTORS_PER_REV))
//=====
int16 sa_EvalDeltaS (uint16 S_Head, uint16 D_Head, uint16 u16_Sector, int32 i32_Cyl)
{
    static int32 i32_DeltaL = 0; // output before final RightShift
    #define C0_SHIFT 15
        // RightShift for total to convert to integer tracks (=15+Qvalue of DC terms))
    #define C1_SHIFT 4
        // RightShift for cylinder input, selected so that:
        //      (Max_Cyl >>C1_SHIFT) is between 8K and 32K.

    i32_DeltaL = ((int32)(Table.i16_DeltaS[D_Head][4] - Table.i16_DeltaS[S_Head][4])
        * (int32)pd_Opr_ai16Q15SineTable[u16_Sector+SECTORS_PER_REV/4 ] ) ;

    i32_DeltaL +=((Table.i16_DeltaS[D_Head][3] - Table.i16_DeltaS[S_Head][3])
        * (int32)pd_Opr_ai16Q15SineTable[u16_Sector]) ;

    i32_DeltaL +=((Table.i16_DeltaS[D_Head][2] - Table.i16_DeltaS[S_Head][2])
        * (i32_Cyl >> C1_SHIFT)*((i32_Cyl >> C1_SHIFT)>>C0_SHIFT) ) ;

    i32_DeltaL +=((Table.i16_DeltaS[D_Head][1] - Table.i16_DeltaS[S_Head][1])
        * (i32_Cyl >> C1_SHIFT)) ;

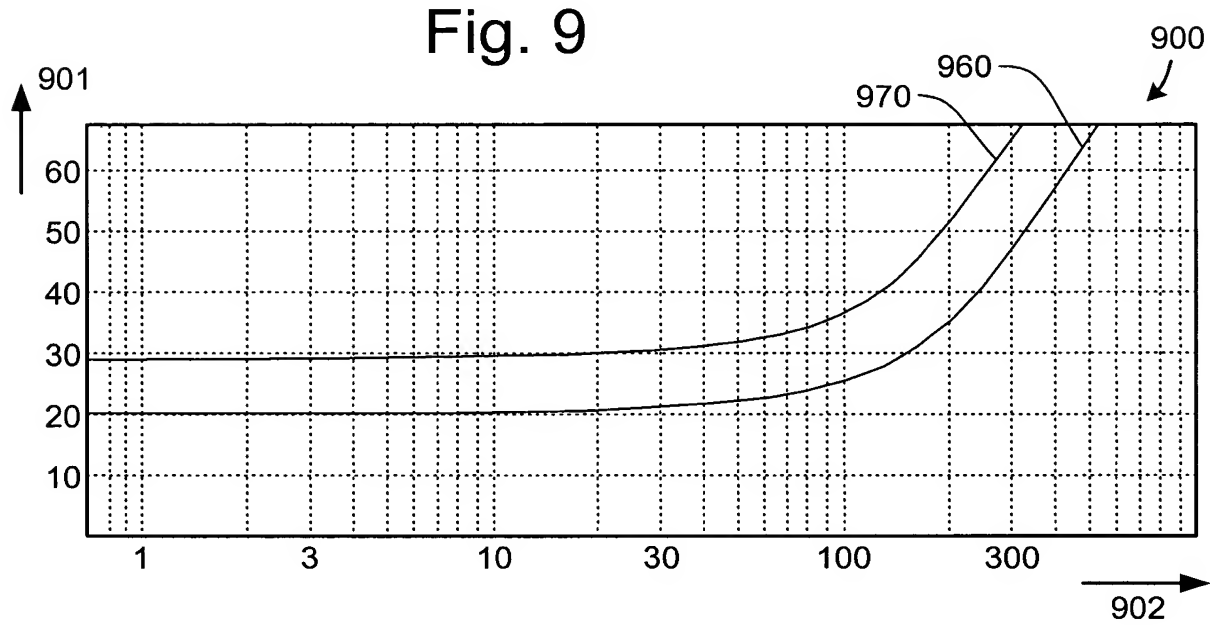
    i32_DeltaL +=(Table.i16_DeltaS[D_Head][0] - Table.i16_DeltaS[S_Head][0])
        << C0_SHIFT; // could also add this term after final shift

    return (int16) ((i32_DeltaL + (1<<(C0_SHIFT-1) ))>> C0_SHIFT);
    // round off and shift to q0.
}

```

	880	870	871	872	873	874	
810	0	16	43	-1775	-32	-10	800
811	1	16	32	-1288	-31	-11	
812	2	-13	402	248	-2	0	
813	3	0	0	0	0	0	

Fig. 8



	1051	1052	1053	1054		1061	1062	1063	1064
1001	0	18129	0	1	...	-12871		-12871.0	
1002	0	31045	0	6	...	45	-29.7	15.3	
	0	30994	3	13	...	-6	-10.0	-16.0	
	0	30953	3	16	...	-47	-10.0	-57.0	
	0	26514	3	28	...	-4486		-4486.0	
	0	26514	2	33	...	-4486		-4486.0	
	1	18124	2	33	...	-12876		-12876.0	
1008	0	30969	2	44	...	-31	0.0	-31.0	
1009	0	26528	1	56	...	-4472		-4472.0	N
1010	1	30998	2	59	...	-2	0.0	-2.0	N
1011	0	31014	2	63	...	14	0.0	14.0	N
1012	0	14733	3	65	...	-16267		-16267.0	N
1013	0	30973	0	69	...	-27	-29.6	-56.6	N
1014	0	31022	1	78	...	22	-28.3	-6.3	N
1015	1	31039	1	81	...	39	-28.4	10.6	Y
	0	31029	1	88	...	29	-28.3	0.7	Y
	0	26523	1	96	...	-4477		-4477.0	N
	0	31010	3	97	...	10	-10.0	0.0	Y
	1	26514	1	106	...	-4486		-4486.0	N
	1	31026	2	108	...	26	0.0	26.0	Y
	0	18098	0	111	...	-12902		-12902.0	N
	0	31020	2	114	...	20	0.0	20.0	Y
	0	18125	2	116	...	-12875		-12875.0	N
	0	31024	3	117	...	24	-10.0	14.0	Y
	0	18122	2	117	...	-12878		-12878.0	N
	0	26520	1	119	...	-4480		-4480.0	
	0	26526	0	122	...	-4474		-4474.0	
	0	31033	1	127	...	33	-28.3	4.7	
	0	31027	3	130	...	27	-10.0	17.0	
	1	31022	0	142	...	22	-29.6	-7.6	
	0	18124	2	145	...	-12876		-12876.0	
	1	30989	1	157	...	-11	-28.3	-39.3	
	1	31052	2	164	...	52	0.0	52.0	
	0	30997	2	169	...	-3	0.0	-3.0	
	0	30958	3	172	...	-42	-10.0	-52.0	
	1	2801	3	184	...	-28199		-28199.0	
	0	15275	1	188	...	-15725		-15725.0	
1038	0	31043	3	189	...	43	-10.0	33.0	
	1000				1060

Fig. 10